

Examenul național de bacalaureat 2021
Proba E. d)
INFORMATICĂ
Limbajul C/C++

Testul 10

Filieră teoretică, profil real, specializare științe ale naturii

- Toate subiectele sunt obligatorii. Se acordă zece puncte din oficiu.
- Timpul de lucru efectiv este de trei ore.
- Identificatorii utilizați în rezolvări trebuie să respecte precizările din enunț (**bold**), iar în lipsa unor precizări explicite, notațiile trebuie să corespundă cu semnificațiile asociate acestora (eventual în formă prescurtată). Datele de intrare se consideră corecte, validarea lor nefiind necesară.

I. TÉTEL

(20 punct)

Az 1-től 5-ig számozott itemek esetén írja a vizsgalapra a helyes válasz betűjelét! Minden helyes válasz 4 pontot ér.

1. Az alábbi C/C++ kifejezés
`!(a<2001) && (a<=2002) || (a>=2020) && !(a>2021)`
értéke akkor és csak akkor 1, ha az **a** változóban tárolt valós szám a következő halmaz eleme:
a. $[2001, 2002] \cap \{2020, 2021\}$ b. $[2001, 2002] \cup [2020, 2021]$
c. $\{2001\} \cup [2002, 2020] \cup \{2021\}$ d. $[2001, 2002] \cap \{2020\} \cap \{2021\}$
2. Az **x** és **y** változók valós típusúak. Adja meg az **x** és **y** változók értékét a mellékelt utasítássor végrehajtása után!
`x=10.23; y=5.14;
if (floor(x)==floor(y)) x=y;
else x=x+y;
x=floor(x); y=floor(y);`
a. **x=5** és **y=5** b. **x=10** és **y=15** c. **x=15** és **y=10** d. **x=15** és **y=5**
3. Az **x** egydimenziós tömb elemei (5, 9, 10, 13, 19). Az **x** tömbnek az **y** tömbbel való növekvő sorrendben történő összefésülése során kapott tömb elemei (5, 8, 9, 10, 12, 13, 19, 19, 30, 52). Adja meg az **y** tömb elemeit abban a sorrendben, ahogyan ebben előfordulnak!
a. (52, 30, 19, 12, 8) b. (36, 20, 12, 6, 6)
c. (5, 9, 10, 13, 19, 8, 12, 19, 30, 52) d. (5, 6, 9, 6, 10, 12, 13, 20, 19, 36)
4. Annak ellenőrzésére, hogy az **x=16** értékű elem szerepel-e a (2, 5, 8, 11, 16, 22, 40) egydimenziós tömbben, a bináris keresés módszerét alkalmazzuk. Adja meg azoknak az elemeknek a sorozatát, amelyeket a módszer alkalmazása során rendre összehasonlítottunk az **x**-szel!
a. 11, 16 b. 16 c. 11, 22, 16 d. 22, 16
5. Adja meg azt az utasítássorozatot, amely elvégzi az **x** és **y** egész típusú változók értékének felcserélését!
a. `x = x - y;` b. `x = x * y;` c. `x = x + y;` d. `x = x + y;`
`y = x + y;` `y = x / y;` `y = x - y;` `y = x * y;`
`x = x + y;` `x = x / y;` `x = x * y;` `x = x / y;`

II. TÉTEL

(40 pont)

1. **Adott a mellékelt pszeudokód algoritmus:**
Az **x%y** az **x** természetes számnak az **y** nullától különböző természetes számmal való osztási maradékát jelöli.
a. Adja meg az algoritmus által kiírt értéket, ha a beolvasott számok sorrendben a 2 és 9. (6p.)
b. Ha az **x** változóba a 3-as értéket olvassuk be, adjon meg két olyan számot az [1, 20] intervallumból, amit az **y** változóba beolvasva, az algoritmus végrehajtása után mindkét esetben az eredmény pontosan 12 karakterből álljon! (6p.)
c. Írja meg az algoritmusnak megfelelő C/C++ programot! (10p.)
d. Írjon az adott algoritmussal egyenértékű pszeudokód algoritmust, amelyben az **amíg ... végezd el** ismétlő szerkezetet, egy más típusú ismétlő szerkezettel helyettesíti! (6p.)

```
beolvas x, y (természetes számok)
ha x < y akkor
    x ← x - y; y ← x + y; x ← y - x
amíg x ≥ y végezd el
    kiír 'A'
    x ← x - y
    ha x % 2 = 0 akkor kiír 'A'
    különben kiír 'B'
```

2. A `cod` változóban egy diák vizsgaazonosító kódját (egy természetes szám az $[1, 10^3)$ intervallumból), a `nota1` és `nota2` változókbán pedig a diák vizsgajegyeit tároljuk (valós számok). Deklarálja a `nota1` és `nota2` változókat, majd írjon egy olyan C/C++ utasítássort, amelynek végrehajtása után a képernyőn a diák vizsgakódja lesz kiírva, majd a képernyő következő sorában a két megadott vizsgajegy átlaga! (6p.)
3. Írja le, hogy mi fog megjelenni a képernyőn az alábbi utasítássor végrehajtása után, tudva azt, hogy mindegyik változó `char` típusú!
- ```
c2='o'; c1=c2+1; c3=c2-2;
cout<<c1<<c2<<c3<<endl; | printf("%c%c%c\n", c1, c2, c3);
c3=c1+2; c2='e'; c1=c3+4; c4=c2-1;
cout<<c1<<c2<<c3<<c4<<c2; | printf("%c%c%c%c%c", c1, c2, c3, c4, c2);
```
- (6p.)

### III. TÉTEL

(30 pont)

1. Az `x` és `y` változók **harmonikusak**, ha az összegük az `x` osztóinak összege és az `y` osztóinak összege által meghatározott nyílt intervallum eleme.  
Olvasson be két nemnulla természetes számot, az `x` és `y`-t, majd írja ki az 1-es értéket, ha az `x` és `y` változók harmonikusak, illetve a 0 értéket, máskülönben!  
Írjon pszeudokód algoritmust a feladat megoldására!  
**Példa:** ha `x=8` és `y=12`, akkor 1 lesz kiírva ( $1+2+4+8=15$ ,  $1+2+4+6+12=25$ , és  $8+12=20 \in (15, 25)$ ), ha `x=8` és `y=13`, akkor 0 lesz kiírva ( $1+2+4+8=15$ ,  $1+13=14$ , és  $8+13=21 \notin (14, 15)$ ). (10p.)
2. Egy érték egy sorozatot **szűr**, ha van a sorozatnak az adott számmal megegyező értékű két olyan eleme, melyek közül az egyik elem a sorozat első felében, míg a másik a sorozat második felében található.  
Írjon C/C++ programot, amely természetes számokat olvas be a billentyűzetről a  $[2, 20]$  intervallumból: az `n`-et és  $2 \cdot n$  számot, amelyek egy egydimenziós tömb elemeit képezik, azzal a tulajdonsággal, hogy az első `n` darab szám is különböző, illetve a második `n` darab szám is különböző a tömbben! A program tetszőleges sorrendben írja ki azokat az értékeket, amelyek szűrhetik a sorozatot, egy-egy szóközzel elválasztva, vagy a `nu exista` üzenetet, ha nincsenek ilyen értékek a sorozatban!  
**Példa:** ha `n=4` és a tömb  $(\underline{4}, 5, \underline{7}, \underline{2}, \underline{2}, 6, \underline{4}, \underline{7})$ , akkor a képernyőre kiírt értékek a 2 4 7 számok, nem feltétlenül ebben a sorrendben. (10p.)
3. A `bac.txt` szöveges állomány legtöbb  $10^5$  természetes számot tartalmaz az  $[1, 10^9]$  intervallumból, szóközzel elválasztva.  
Írassa ki a képernyőre azt a legkisebb pozíciót, amelyet az állomány első eleme elfoglalna, ha növekvő sorrendbe rendeznénk a számsort! Tervezzen memória és idő szempontjából hatékony algoritmust!  
**Példa:** ha az állomány a következő értékeket tartalmazza: 15 7 16 17 6 4  
akkor a képernyőre kiírt érték a 4 lesz.  
(15 a negyedik helyen áll a sorozatban 4, 6, 7, 15, 16, 17).  
a. Írja le a saját szavaival a tervezett algoritmust, megindokolva annak hatékonyságát! (2p.)  
b. Írja meg a megtervezett algoritmusnak megfelelő C/C++ programot! (8p.)